



0/ 1302068264US

A/NO
603597 Fee

ACT-185

This application is submitted in the names of inventors John McGowan, William Plants, Joel Landry, Sinan Kaptanoglu, and Warren Miller assignors to Actel Corporation, a California corporation.

SPECIFICATION

5

FLEXIBLE, HIGH-PERFORMANCE STATIC RAM ARCHITECTURE FOR FIELD-PROGRAMMABLE GATE ARRAYS

BACKGROUND OF THE INVENTION

1. Field of the Invention

The present invention relates to field programmable gate array (FPGA) integrated circuits. More particularly, the present invention relates to FPGA integrated circuits including static random access memory devices within the array of logic modules.

2. The Prior Art

As integrated circuit technology advances, geometries shrink, performance improves, and densities increase. This is especially true in logic products such as Application Specific Integrated Circuits (ASICs), Complex Programmable Logic Devices (CPLDs), and Field Programmable Gate Arrays (FPGAs). This trend makes the design of systems of ever increasing complexity at ever decreasing cost feasible. One of the requirements of these systems is fast, flexible, inexpensive memory for a variety of purposes such as register files, FIFOs, scratch pads, look-up tables, etc.

There are significant cost and performance savings to be obtained by integrating this functionality directly into these types of logic products.

Using external SRAMs with FPGA designs is undesirable for several reasons. Separate memory chips are expensive, require additional printed circuit board space, and consume I/O pins on the FPGA itself. Also, a separate memory chip is required to implement each memory function, thereby further increasing the cost.

These difficulties have resulted in various attempts by FPGA manufacturers to include on-chip SRAM resources on their devices. However, such attempts have been less than desirable with regards to cost, performance, and flexibility.

One such attempt is to simply build the SRAM out of array logic and routing resources, using the available logic blocks as gates and latches and using programmable interconnect to connect them. This is extremely costly and slow because it offers no density improvement over ordinary FPGA functionality, consumes a considerable amount of logic array resources, and the critical paths are quite long for even a small memory block.

A variation on this theme (Xilinx 4000 Series) is available on SRAM based FPGAs where the configuration information for the logic blocks and programmable interconnect is stored in SRAM cells. Some of these SRAM cells are used by

configuring the logic blocks as small (16 bit) SRAM blocks. While this distributed SRAM approach is an improvement in density and is flexible for building larger memories, it is still slow and consumes logic array resources. The necessary overhead circuitry was sufficiently large that Xilinx actually removed it when they
5 developed their low cost 4000-D parts.

Another approach recently announced by Altera is to put dedicated memory blocks on board the FPGAs. This has been used to produce large (2K bit), dense, flexible SRAMs with very poor performance. These dedicated memory blocks are exceedingly slow (25ns read access for an on-chip 2K CMOS memory). These
10 memory blocks are single ported, which, while good for density, negatively impacts the speed of some memory functions like FIFOs and register files even more. Further, these memory blocks are limited in extent by the programmable interconnect channels (the interconnect density may exceed that of the rest of the array, thus hindering routeability), and are overly flexible (having too many options hurts speed and
15 routeability).

Another approach to SRAM memory in FPGA applications is found in "Architecture of Centralized Field-Configurable Memory", Steven J. E. Wilton, et. al., from the minutes of the 1995 FPGA Symposium, p. 97. This approach involves a large centralized memory which can be incorporated into FPGA. The centralized memory
20 comprises several SRAM arrays which have programmable local routing interconnect

which are used exclusively by the centralized memory block. The local routing interconnects are used to make efficient the configuration of the SRAMs within the centralized memory block.

Clearly there is a need for an SRAM architecture indigenous to FPGA logic arrays which will provide high performance, density approaching the inherent SRAM density of the semiconductor process, reasonable flexibility, and routing density comparable to the rest of the logic array. Such an architecture would share some of the characteristics of the distributed and dedicated block SRAMs reported in the prior art while incorporating additional characteristics further optimizing it for use in FPGA logic arrays.

BRIEF DESCRIPTION OF THE INVENTION

The presently preferred embodiment of the invention comprises a flexible, high-performance memory integrated into an FPGA architecture. A given FPGA integrated circuit includes a plurality of independent RAM blocks, the number of which is based on the size of the FPGA array. According to a presently preferred embodiment of the invention, each integrated circuit may include from eight to fourteen RAM blocks, depending on the size of the array. Each block contains 256 bits of RAM arranged, for example, as 32x8 or 64x4, and is fully independent from the other blocks.

Connections are made to a block using antifuse connections to horizontal metal

routing channels in the same way that connections are made to logic modules. In accordance with one feature of the present invention, SRAM blocks span more than one logic module row. According to an exemplary actual layout of an architecture according to the present invention, the SRAM block differs from the logic modules in that an SRAM block spans four module rows. Thus, a block is associated with five routing channels.

The SRAM blocks are preferably placed into two dedicated SRAM columns, at intermediate locations in the array that are optimal for automated place-and-route algorithms. Neighboring logic modules can be used in conjunction with the SRAM to produce depth and/or width expansion.

The aforementioned horizontal routing channels pass through the SRAM block allowing logic modules on either side to connect to each other as if the SRAM block were not there. This is quite different from the distributed or dedicated prior-art SRAM included on FPGA integrated circuits because the SRAM block extents are not bounded or limited by the routing channels and the routing channels are not interrupted by the SRAM blocks. In addition, the inputs and outputs to the memory block are distributed amongst the five routing channels in order to mimic the routing density of the logic array as a whole. This feature of the present invention is crucial to maintaining routeability, since if the density of the signals into and out of the SRAM blocks were too high, it would create blockages in the routing channels which could

6

make it difficult or impossible for the different parts of the array to connect, severely limiting the useability of the FPGA. There are also a variety of features included in the structure of the SRAM block itself which facilitate its use inside an FPGA logic array.

BRIEF DESCRIPTION OF THE DRAWING FIGURES

5 FIG. 1 is a block diagram of an FPGA array with dedicated SRAM blocks according to the present invention.

FIG 2. is a more detailed block diagram of a portion of the FPGA array of FIG. 1, showing more detail of the manner in which a typical SRAM block fits into the interconnection scheme of the architecture.

10 FIG. 3 is a block diagram of a typical SRAM block suitable for use in the architecture of the present invention.

FIG. 4 is a timing diagram showing the timing of the write operation of the SRAM block of FIG. 3.

FIG. 5 is a simplified timing diagram showing the typical complex write
15 operation of a level-sensitive commercial SRAM integrated circuit, in contrast to the simple timing of the present invention shown the SRAM block of FIG. 4.

DETAILED DESCRIPTION OF A PREFERRED EMBODIMENT

Those of ordinary skill in the art will realize that the following description of the present invention is illustrative only and not in any way limiting. Other embodiments of the invention will readily suggest themselves to such skilled persons.

5 The presently preferred embodiment of the invention comprises a flexible, high-performance memory integrated into an FPGA architecture. A block diagram of a typical architecture according to the present invention is presented in FIG. 1. FPGA architecture 10 includes a plurality of logic function modules 12 (shown as a plurality of squares, an exemplary one of which is designated by reference numeral 12)

10 arranged in an array comprising a plurality of rows and columns. Those of ordinary skill in the art will readily recognize that the terms "row" and "column" used in both the specification and the claims are interchangeable and equivalent; merely rotating the array 90° converts a row into a column. Arrays formed according to the present invention may or may not be symmetrical in the row and column dimensions.

15 Logic function modules 12 may be any one of a variety of circuits, including, for example, the logic modules disclosed in United States Patent Nos: 4,758,745; 4,873,459; 4,910,417; 5,015,885; 5,451,887 and 5,477,165 assigned to the same assignee as the present invention.

As shown in FIG. 1, selected ones of the logic function modules 12 are

hardwired to peripheral I/O circuitry (an exemplary one of which is designated by reference numeral 14) although those of ordinary skill in the art will recognize that this is not necessary. Such I/O circuitry, used to transport signals onto and off of the integrated circuit containing the FPGA array, is known in the art. Details of such I/O

5 circuitry are not provided herein to avoid unnecessarily complicating the disclosure and obscuring the present invention. Alternately, ~~as is also known in the art,~~

A peripheral I/O circuitry could be connectable to the interior of the array by using interconnect conductors as is known in the art.

According to the present invention, a plurality of SRAM blocks 16 are disposed

10 in the array along with the logic function modules 12. In the illustrative embodiment depicted in FIG. 1, two columns of six SRAM blocks are disposed in the array. Those of ordinary skill in the art will recognize that FIG. 1 is only illustrative, and a requirement that SRAM blocks 16 span entire columns according to the present invention is not to be implied. Such skilled persons will also realize that, while the

15 instant disclosure is made in terms of SRAM blocks spanning columns, the concept disclosed and claimed herein applies equally to such SRAM blocks spanning rows.

For the size of SRAM block employed in the array of FIG. 1, each SRAM block 16 spans the height of four logic function modules 12. The SRAM blocks 16 in the illustrative embodiment of FIG. 1 are placed into two dedicated SRAM columns, at

20 intermediate locations in the array. For any given array, persons skilled in the art may

9

choose locations which are optimal for automated place-and-route algorithms.

Neighboring logic function modules 12 can be used in conjunction with the SRAM blocks 16 to produce depth and/or width expansion.

Absolute numbers of elements included in the architecture of the present invention is not critical. Thus, a given FPGA integrated circuit may include, for example, eight to fourteen independent SRAM blocks, the number of which is based on the size of the FPGA array. According to a presently preferred embodiment of the invention, each block contains 256 bits of SRAM configured, for example, as 32x8 or 64x4, and is fully independent from the other blocks. Persons of ordinary skill in the art will recognize that other numbers of RAM blocks may be utilized.

Referring now to FIG. 2, a more detailed block diagram of a portion of the array of FIG. 1 shows the interconnectivity between the SRAM blocks 16 and the logic function modules 12. FIG. 2 illustrates how connections are made to and from each SRAM block 16 using user-programmable interconnect elements to make selective connections to individual metal interconnect conductors disposed in routing channels in the same way that connections are made between logic function modules 12.

In FIG. 2, an exemplary SRAM block 16 is shown broken up into four segments 16-1, 16-2, 16-3, and 16-4 to illustrate the distribution of its inputs and outputs into the interconnect architecture of the array of the present invention.

According to an exemplary actual layout of an architecture within the scope of the present invention, the SRAM blocks 16 differ in size from the logic function modules 12 in that an SRAM block 16 spans four module rows (as shown in FIG. 1). In FIG. 2, there are four logic function modules 12-1, 12-2, 12-3, and 12-4 located to the left of SRAM block segments 16-1, 16-2, 16-3, and 16-4, and four logic function modules 12-5, 12-6, 12-7, and 12-8 located to the right of SRAM block segments 16-1, 16-2, 16-3, and 16-4. Thus, each SRAM block 16 is associated with five routing channels (numbered 18-1, 18-2, 18-3, 18-4, and 18-5) which are associated with the four rows of logic function modules proximately located to the SRAM block 16. As shown in FIG. 2, each of the five routing channels comprises four interconnect conductors, labeled a, b, c, and d in each routing channel. Where individual conductors are mentioned herein, they will be identified accordingly (e.g., 18-1b, 18-3a, etc.) Persons of ordinary skill in the art will understand that the use of four conductors are merely illustrative and that the number four was chosen to both illustrate the invention and avoid over complicating the drawing figure which would unnecessarily obscure the disclosure.

Those of ordinary skill in the art will also recognize that the particular arrangement of size and span of the distributed SRAM block 16 comprising SRAM block segments 16-1, 16-2, 16-3, and 16-4 shown in the illustrative embodiment of the present invention does not limit the invention to the disclosed embodiment. Such

skilled persons will readily appreciate that other sizes of SRAM memory blocks may be employed without departing from the concepts of the present invention.

The horizontal routing channels 18-1, 18-2, 18-3, 18-4, and 18-5 are associated with the SRAM block segments 16-1, 16-2, 16-3, and 16-4. The horizontal routing channels 18-2, 18-3, and 18-4 pass through SRAM block segments 16-1, 16-2, 16-3, and 16-4, and horizontal routing channels 18-1 and 18-5 pass between adjacent SRAM blocks 16 in the SRAM column. This allows logic modules on either side to connect to each other as if the SRAM block 16 comprising SRAM block segments 16-1, 16-2, 16-3, and 16-4 was not there. This is quite different from the distributed or dedicated prior-art SRAM included on FPGA integrated circuits because the SRAM block extents are not bounded or limited by the routing channels and the routing channels are not interrupted by the SRAM blocks 16. In addition, the inputs and outputs to the memory block are distributed amongst the five routing channels in order to mimic the routing density of the logic array as a whole.

This feature of the present invention is crucial to maintaining routeability, since if the density of the signals into and out of the SRAM blocks 16 were too high, it would create blockages in the routing channels which could make it difficult or impossible for the different parts of the array to connect, severely limiting the useability of the FPGA. There are also a variety of features included in the structure of the SRAM block 16 itself which facilitate its use inside an FPGA logic array. The distribution of SRAM

block 16 inputs and outputs according to the present invention will now be disclosed in more detail.

Segments 16-1

A The address and data inputs of SRAM block ~~segments 16-1~~, 16-2, 16-3, and 16-4 are shown as input inverters for ease of illustration. Four illustrative inputs 1, 2, 3, and 4 (which may be either control, data or address inputs) are shown for each of SRAM block segments 16-1, 16-2, 16-3, and 16-4, but those of ordinary skill in the art will understand that the total number of control, address and data inputs employed in any actual implementation of the present invention will vary and will be dictated by the width of a data word in the SRAM and the number of address locations needed.

10 As shown in FIG. 2, the inputs of the SRAM block 16 are distributed among the four SRAM block segments 16-1, 16-2, 16-3, and 16-4 in order to optimize routability. Each input conductor intersects the interconnect conductors in one of the wiring channels 18-1, 18-2, 18-3, 18-4, and 18-5. User-programmable interconnect elements are provided at some or all of the intersections. Such interconnect elements may be
15 antifuses, pass transistors controlled by RAM cells, non-volatile memory cells, etc., all of which are well known in the art. These user-programmable interconnect elements are not shown in FIG. 2 due to space limitations. In addition, the outputs (two illustrative outputs labeled 5 and 6 are shown for each SRAM block segment 16-1, 16-2, 16-3, and 16-4) of the SRAM block 16 are distributed among the wiring channels
20 18-1, 18-2, 18-3, 18-4, and 18-5. In the embodiment shown in FIG. 2, each output

13

conductor spans the individual interconnect conductors of four wiring channels, two above, and two below the output. Thus, the outputs from SRAM block segment 16-1 intersect the four interconnect conductors of wiring channels 18-1, 18-2, and 18-3, as well as continuing on to a wiring channel located above the top of the drawing figure.

5 Similarly, the outputs from SRAM block segment 16-2 intersect the four interconnect conductors of wiring channels 18-1, 18-2, 18-3, and 18-4; the outputs from SRAM block segment 16-3 intersect the four interconnect conductors of wiring channels 18-2, 18-3, 18-4, and 18-5; and the outputs from SRAM block segment 16-4 intersect the four interconnect conductors of wiring channels 18-3, 18-4, and 18-5, as well as
10 continuing on to a wiring channel located below the bottom of the drawing figure. Those of ordinary skill in the art will recognize that each output conductor could span a number of wiring channels other than four and could also be programmably connected to other interconnect resources, such as longer lines running all or most of a row or column dimension of the array.

15 The distribution of the inputs and outputs of the SRAM block segments 16-1, 16-2, 16-3, and 16-4 and the pass of the wiring channels through the SRAM blocks 16 allow for optimum interconnect flexibility. The various aspects of this flexibility are illustrated in FIG. 2.

First, the output 6 of SRAM block segment 16-1 is shown connected to an
20 illustrative input of logic function module 12-5 in the same row and to the right of the

14

SRAM block segment 16-1. The two programmed user-programmable interconnect elements (one at the intersection of output 6 of SRAM block segment 16-1 and interconnect conductor 18-2c and the other at the intersection of interconnect conductor 18-2c and the illustrative input conductor of logic function module 12-5) are each represented by an "X" at the appropriate intersection. In addition, output 5 of SRAM block segment 16-1 is shown driving a signal onto interconnect conductor 18-1c. This signal will be used by a module located in another portion of the array not shown in FIG. 2.

Two of the illustrative inputs 1 and 2 of SRAM block segment 16-2 are shown connected to interconnect conductors 18-3a and 18-3b, respectively. As implied by FIG. 2, the sources of the driving signals for inputs 1 and 2 of SRAM block segment 16-2 are located in a portion of the array not illustrated in FIG. 2 and are routed through the general interconnect architecture.

In the third row, an illustrative output of logic function module 12-7, located to the right of SRAM block segment 16-3, is shown driving an illustrative input of logic function module 12-3, located to the right of SRAM block segment 16-3. This is a powerful feature of the present invention, since it permits interconnection of logic function modules located adjacent to the SRAM blocks 16 as if the SRAM blocks 16 were not present, thus rendering the SRAM blocks 16 virtually transparent to the routing resource.

Finally, in the fourth row of the array illustrated in FIG. 2, input 1 of SRAM block segment 16-4 is shown being driven by an illustrative output of logic function module 12-8 via interconnect conductor 18-5a and two programmed user-programmable interconnect elements, while inputs 2, 3, and 4 of SRAM block segment 16-4 are
5 shown being driven from signals on interconnect conductors 18-5b, 18-5c, and 18-5d which have come from other locations in the array.

The block diagram of a single SRAM block 16 comprising SRAM block segments 16-1 through 16-4 according to a presently preferred embodiment of the invention is shown in FIG. 3. The SRAM block 16 is preferably dual-ported, permitting
10 simultaneous writes and reads from different addresses. As shown in FIG. 3, the size of the memory is 256 bits, which can preferably be configured with either of two data word widths, for example as 32 8-bit bytes or 64 4-bit nibbles. Those of ordinary skill in the art will readily recognize that the memory size may be other than 256 bits and that the data word width may be other than 8-bit bytes or 4-bit nibbles. The
15 architecture of SRAM components is well known, and persons of ordinary skill in the art will be readily able to design SRAM blocks 16 such as illustrated in FIG. 3 from individual transistors.

Dual-porting the SRAM blocks 16 of the present invention is important for attaining high performance, since it allows the use of a current sensing read scheme

16

which is much faster than the traditional differential voltage sense amplifier used in most single port SRAM designs. Separating the write port from the read port eliminates write recovery times from the read access path, which further enhances the speed. There are several disadvantages of dual porting the SRAM blocks 16 of the preferred embodiment of the present invention. They include the additional decode circuitry required and the additional address lines which increase the routing density around the SRAM block ^{Segments 16-1} ~~segments 16-1~~ through 16-4. The presence of the additional address lines is compensated for by spreading the SRAM block 16 inputs over several routing channels as previously described herein. Reducing the number of data word width modes to two (byte-wide or nibble-wide data words) allows the user some configuration flexibility without significantly increasing control circuitry or harming access time.

As previously disclosed, in its preferred embodiment, the SRAM block 16 of the present invention is distributed over an area normally occupied by four logic function modules in an FPGA array. It has been found optimal to distribute the components of the SRAM block among the four portions 16-1, 16-2, 16-3 and 16-4 in as linear a manner as is practical in any given FPGA layout.

The architecture of the SRAM blocks 16 used in a preferred embodiment of the present invention includes a RAM array 20 communicating with write word select circuit 22 and read word select circuit 24. A row of bit line drivers 26 take the write

data from write latches 28 driven by an eight bit write data (WD) bus 30. As shown in FIG. 3, interconnect conductors from wiring channels 18-n are shown intersecting write data bus 30 and are connectable thereto by user-programmable interconnect elements 32 (shown as circles).

5 The data in write latches 28 is written into an address in RAM array 20 selected by write word select circuit 22 and bit line drivers 26 from the address data present on a 6-bit write address (WRAD) bus 36 which has been latched by write address latch 34. Interconnect conductors from wiring channels 18-n are shown intersecting write address bus 36 and are connectable thereto by user-programmable interconnect
10 elements 32 (shown as circles).

 The write operations are controlled by write logic circuit 38 in accordance with its control inputs including MODE control input 40, block enable (BLKEN) input 42, write enable (WEN) input 44, and write clock (WCLK) input 46. The MODE control input 40 sets the byte/nibble setting of the memory address location widths by
15 programming it to the appropriate logic level.

 Use of the nibble mode by activating MODE control input 40 requires an additional address line on both ports but reduces the number of data lines by four (4) at each port. The net savings is six signals per SRAM block 16. As numerous SRAM blocks 16 occupy a given column in the presently preferred embodiment of the

invention, and since they utilize a common group of vertical routing resources associated with the column, use of the nibble mode can reduce the probability of running out of such resources during automated place and route. Additionally, use of nibble mode may permit automated place and route to choose a preferred set of routing resources which may offer improved speed of operation. Those of ordinary skill
5 in the art will recognize that this provides maximum flexibility and performance.

Input data (WD) on bus 30, write address (WRAD) data on bus 36, and control signals (WEN input 44 and BLKEN input 42) are synchronized to write clock (WCLK)
46. The polarity of the WCLK 46 may selectable by placing a logic 0 or logic 1 at write
10 clock polarity input (WCLKP) 48. As will be appreciated by those of ordinary skill in the art, this may be easily accomplished by feeding the WCLK input 46 to one input of an exclusive-OR gate and tying the WCLKP input 48 to the other input of the gate to logic 0 or logic 1 as is well known in the art. A write operation takes place on the appropriate edge of WCLK input 46 whenever WEN input 44 and BLKEN input 42 are
15 both logic HIGH. The BLKEN input 42, like WCLK input 46, may employ programmable polarity selection as described above using BLKENP input 50.

It will be appreciated that the inputs to write logic circuit 38 are connectable to Interconnect conductors in a wiring channel by user-programmable interconnect elements as depicted in FIG. 2. To avoid over complicating drawing FIG. 3 and
20 unnecessarily obscuring the disclosure, the user-programmable interconnections

19

between the inputs to the write logic circuit 38 and the interconnect conductors are not shown in FIG. 3.

As will be appreciated by those of ordinary skill in the art, programmably selecting the polarity of the BLKEN input 42 to the SRAM block 16 allows two different
5 SRAM blocks 16 programmed with opposite polarity on their BLKEN inputs 42 to effectively have a common seventh address bit. This saves the user from expending modules and routing resources to implement this. The user can still use WEN=0 to disable both blocks.

Synchronizing the timing of the write port to the WCLK input 46 is important
10 because it simplifies the timing for the user. The illustrative write timing for the user of the SRAM block 16 of the present invention is shown in the timing diagram of FIG. 4. All memories (even asynchronous ones) have write timing requirements because address, data, and control signals must be held constant during the duration of a write pulse or false data may be written into the SRAM array. A synchronous write port
15 moves all of the complicated timing relationships, such as the ones normally encountered in SRAM devices of this type and illustrated in FIG. 5, inside the SRAM block 16 relieving the user of the burden of generating a number of timed pulses. Providing polarity select on various control signals as described above allows the user further flexibility in both the logic design and the use of multiple SRAM blocks 16 to
20 construct deeper or wider memories. This little bit of logic can save a considerable

amount of logic array resources and helps relieve routing density bottlenecks.

With the synchronous timing of the write port, the latches 28 and 34 each perform as masters to a common slave comprising the write word select circuit 22, the bit line drivers 26 and the selected storage elements of the RAM array 20. This gives
5 the write operation the appearance of simply clocking the data into a D-flip/flop on the active edge of WCLK 46 as illustrated in FIG. 4. Both of the latches 28 and 34 are alternately transparent and latched on opposite phases of the clock. When WCLK input 46 is LOW, latches 28 and 34 are transparent, data is presented to the inputs of the bit line drivers 26 and the location of the data to be written is presented to the
10 inputs of the write word select circuitry 22 and the bit line drivers 26. When WCLK input 46 is brought HIGH, the latches 28 and 34 also latch the state of the WRAD 36 and WD 30 busses, the selected bit line drivers drive the data onto the bit lines of RAM array 20, the write word select circuitry 22 selects the word location where the data is to be stored, and the data is written into the now-transparent latches in the selected
15 memory elements in the RAM array 20. When the WCLK is again brought LOW, the previously selected latches in the RAM array 20 latch the data.

The RAM array 20 may be read by placing a read address on read address bus 52. Interconnect conductors from wiring channels 18-n are shown intersecting read address bus 52 and are connectable thereto by user-programmable interconnect
20 elements 32 (shown as circles). The read address may be latched into read address

latch 54. The read address is output from read address latch 54 and presented to read word select circuit 24 to select the the data word to be read from RAM array 20. RAM array 20 drives column multiplexer 56, whose function is to choose select data (data byte) from select memory cells at the memory address selected by read word select circuit 24. The data byte selected by the column multiplexer 56 is output to the sense amplifiers 58 which are driven by the column multiplexer 56. When the SRAM block 16 is in the nibble mode, the nibble multiplexer 60, in response to the mode signal 40 and the address latch 54, further selects data (data nibble) from the data byte being transmitted through the sense amplifiers 58. Otherwise, the nibble multiplexer 60 is transparent. The sense amplifiers 58 drive both the nibble multiplexer 60 and output latches 62 to place the nibble or byte on read data bus 64. Interconnect conductors from wiring channels 18-n are shown intersecting read data bus 64 and are connectable thereto by user-programmable interconnect elements 32 (shown as circles).

The control inputs to read logic 66 on the read side include latch enable (LEN) input 68, read enable (REN) input 70, and read clock (RCLK) 72. On the read side, all eight data outputs on read dat bus 64 will be used for byte mode. For nibble mode only the four lowest-order bits will be connected to other logic via user-programmable interconnect elements. In byte mode the highest order read and write address bits become don't-cares.

According to a presently preferred embodiment of the invention, the read operation may be performed either synchronously or asynchronously. When the read port is synchronous, the read addresses on read address bus 52 and read data on read data bus 64 are synchronized to the RCLK input 72 whenever the output latch enable (LEN) input 68 is programmed to a logic 1. When the read port is
5 asynchronous, the LEN input 68 is programmed LOW and the read address latches 54 and output data latches 62 are forced transparent. In this latter mode, output data will change in response to a change in read address, as opposed to changing in response to an edge on RCLK input 72. As with the WCLK input 46, the RCLK input 72
10 preferably includes programmable polarity using the RCLKP input 74.

Finally, the read enable (REN) control input 70 of SRAM block 16 in the preferred embodiment of the invention implements a power-down feature. When REN=0 the sense amplifiers 58 are powered down, permitting zero standby power. A hold-state latch preserves the previous state of the read data (RD) despite having the
15 sense amplifiers 58 inactive.

It will be appreciated that the inputs to read logic 66 are connectable to Interconnect conductors in a wiring channel by user-programmable interconnect elements as depicted in FIG. 2. To avoid over complicating drawing FIG. 3 and unnecessarily obscuring the disclosure, the user-programmable interconnections
20 between the inputs to the write logic circuit 38 and the interconnect conductors are not

shown in FIG. 3.

Synchronously latching the read address and data signals in SRAM block 16 is important because it allows the user greater flexibility and improved performance. In the synchronous mode, the read address latches 54 are alternately transparent and
5 latched on opposite phases of RCLK 72 and are 180 degrees out of phase relative to the output latches 62. Thus read address latches 54 and output latches 62 perform analogously to the two latches in a master/slave flip/flop. The SRAM block 16 appears to have an internal register allowing pipelined operation (further boosting performance) in high speed systems.

10 The FPGA architecture described herein offers flexible, high-performance SRAM to the user of FPGAs. The flexibility of the architecture permits efficient implementation of on-chip data storage, register files, and FIFOs. Small-capacity high-speed dual-port SRAM can be used to handle ATM data packets; for DRAM and DMA control; as a "rubber-band" synchronizer between two clocks of differing frequency;
15 and as a coefficient table for FIR and IIR filters (wherein many integer coefficients are stored once and retrieved repeatedly).

By offering many independent blocks, the FPGA can support many different sorts of applications. Unused blocks can be turned into 8-bit registers by fixing the write and read addresses and tying all enables HIGH (except LEN which is tied LOW).

On-chip SRAM is many times more efficient for storing data than logic modules and saves many valuable I/O pins. Thus, the user can fit more logic into, and obtain greater performance from, a given FPGA.

Those of ordinary skill in the art will recognize that the SRAM architecture
5 disclosed herein can also be utilized for FIFO, ROM, and as single port RAM with or without employing a bidirectional data bus.

While embodiments and applications of this invention have been shown and described, it would be apparent to those skilled in the art that many more modifications than mentioned above are possible without departing from the inventive concepts
10 herein. The invention, therefore, is not to be restricted except in the spirit of the appended claims.